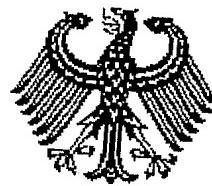


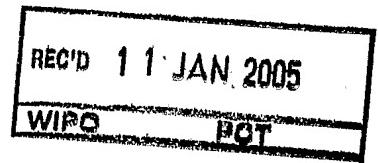
BUNDESREPUBLIK DEUTSCHLAND

**PRIORITY
DOCUMENT**
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)



PCT/EP2004/012687

10.12.2004



Prioritätsbescheinigung über die Einreichung einer Patentanmeldung

Aktenzeichen:

103 57 118.3

Anmeldetag:

6. Dezember 2003

Anmelder/Inhaber:

DaimlerChrysler AG, 70567 Stuttgart/DE

Bezeichnung:

Laden von Software-Modulen

IPC:

G 06 F 13/00

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

München, den 17. November 2004
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag

Stanschus

DaimlerChrysler AG

Dr. Nili

03.12.2003

Laden von Software-Modulen

Die Erfindung betrifft ein Verfahren zum Laden eines Software-Moduls in eine Prozessoreinheit eines Steuergeräts in einem Verkehrsmittel, wobei das Software-Modul in mehreren Steuergeräten lauffähig ist und die Steuergeräte über einen Datenbus Daten austauschen.

In der DE 196 31 309 A1 ist eine Mikroprozessoranordnung für ein Fahrzeug-Regelungssystem mit mehreren, untereinander durch Bussysteme verbundenen Mikroprozessorenensystemen offenbart.

Es ist die Aufgabe der vorliegenden Erfindung, die Prozessorauslastung in miteinander vernetzten Steuergeräten zu optimieren.

Diese Aufgabe wird erfindungsgemäß durch die Merkmale des Anspruchs 1 gelöst. Danach erfolgt die Auswahl auf welchem Steuergerät das Software-Modul geladen wird, in Abhängigkeit der Rechenkapazität der sich aktuell im Betrieb befindlichen Steuergeräte. Mit dem Auswahlverfahren ist sichergestellt, dass das Software-Modul auf dem geladenen Steuergerät aktuell genügend Rechenkapazität zur Abarbeitung seiner Prozesse vorfindet und nicht auf einem Steuergerät gestartet wird, auf dem aktuell nicht genügend Rechenkapazität vorhanden ist. Das Auswahlverfahren erlaubt eine gezielte Ausnutzung von freien Rechenkapazitäten in einem Verbund von Steuergeräten, welche untereinander kommunizieren können.

Bevorzugt wird die Rechenkapazität der Steuergeräte turnusmäßig oder auf Anfrage ermittelt. Dies hat den Vorteil, dass bekannt

ist, welches Steuergerät aktuell über wie viel freie Rechenkapazität verfügt. Mit dieser Information kann entsprechend das Laden des Software-Moduls auf ein bestimmtes Steuergerät gesteuert werden. Die freie Rechenkapazität eines Steuergeräts ist von den aktuell von diesem Steuergerät zu bearbeitenden Aufgaben abhängig ist. Diese ist damit Schwankungen unterworfen und muss den weiteren Steuergeräten mitgeteilt werden.

Vorteilhafterweise wird die Rechenkapazität eines Steuergeräts aus der Prozessorauslastung und dem Prozessortyp ermittelt, so dass sichergestellt wird, dass selbst bei unterschiedlichen Prozessortypen die freie Rechenkapazität korrekt bestimmt wird, insbesondere nicht nur die Prozessorauslastung herangezogen wird.

Vorzugsweise wird das Software-Modul auf dem Steuergerät mit der maximalen freien Rechenkapazität gestartet, so dass nicht Steuergeräte mit weniger Rechenkapazität mit der Ausführung des einen Software-Moduls belastet werden.

Bevorzugt vergleicht das Steuergerät auf dem das Software-Modul läuft seine Rechenkapazität mit der Rechenkapazität der weiteren Steuergeräte. In Abhängigkeit des Vergleichs wird das Software-Modul von dem Steuergerät beendet oder fortgeführt. Dies hat den Vorteil, dass bei Prozessorauslastungsveränderungen an dem Steuergerät das Software-Modul abgeschalten kann.

Vorteilhaft wird bei Beendigung des Software-Moduls ermittelt, welches der weiteren Steuergeräte die maximal freie Rechenkapazität zur Verfügung stellt. Zudem wird das Software-Modul auf diesem Steuergerät gestartet.

Vorteilhafterweise ist das Software-Modul auf den Steuergeräten lauffähig, denn sonst kann das Software-Modul nicht von den Steuergeräten geladen werden. Zudem befinden sich die Steuergeräte im laufenden Betrieb. Das Laden des Software-Moduls erfolgt also zur Laufzeit zumindest des Betriebssystems und mög-

licherweise weiterer Software-Module, welche auf dem betreffenden Steuergerät geladen sind.

Bevorzugt sendet das Software-Modul eine Kennung über seinen Betriebszustand und sein Betriebs-Steuergerät, also eine Kennung des Steuergeräts auf dem das Software-Modul läuft, turnusmäßig oder auf Anfrage auf den Datenbus. Damit ist sichergestellt, dass die korrekte Funktion überprüft sowie die Einflussnahme auf das Software-Modul direkt erfolgen kann.

Es gibt nun verschiedene Möglichkeiten, die Lehre der vorliegenden Erfindung in vorteilhafter Weise auszustalten und weiterzubilden. Dazu sind einerseits auf die untergeordneten Ansprüche und andererseits auf die nachfolgende Erläuterung einer Ausführungsform zu verweisen. Es sollen auch die vorteilhaften Ausgestaltungen einbezogen sein, die sich aus einer beliebigen Kombination der Unteransprüche ergeben. In den Zeichnungen ist eine Ausführungsform des erfindungsgemäßen Verfahrens und eine Vorrichtung dargestellt. Es zeigen jeweils in schematischer Darstellung,

Fig. 1 eine Vorrichtung zur Durchführung des erfindungsgemäßen Verfahrens,

Fig. 2 ein Verfahrensablauf zur Durchführung des erfindungsgemäßen Verfahrens und

Eine Übersicht einer Vorrichtung zur Durchführung des erfindungsgemäßen Verfahrens ist in Figur 1 dargestellt. Die an einem Bussystem eines Verkehrsmittels 9 teilnehmenden Komponenten sind über einen Datenbus 8 miteinander verbunden. Die teilnehmenden Komponenten bestehen vorzugsweise aus Steuergeräten, Sensoren und Aktoren. In der schematischen Figur 1 stellen die Komponenten Steuergeräte 1,3,5 mit entsprechenden auf diesen laufenden Software-Modulen 2,4,6,7 dar.

Mittels der eingesetzten Betriebssysteme können die Steuergeräte 1,3,5 bzw. deren Software-Module 2,4,6,7 untereinander kommunizieren. Dabei bedient man sich Standards, die auf dem Gebiet der Software für Fahrzeuge, bereits etabliert sind. Einige dieser Standards sind das in die ISO 15765-2 übernommene OSEK - Offene Systeme und deren Schnittstellen für die Elektronik im Kraftfahrzeug - (<http://www.osek-vdx.org>), als Transportprotokoll zwischen Steuergeräten oder das in die ISO 14230 (<http://www.iso.org>) übernommene Keyword Protocol 2000 zur Übertragung von Diagnosedaten sowie der Bereitstellung von Diagnosediensten.

Als Kommunikationsprotokoll steht das Keyword Protocol 2000 (KWP 2000) zur Verfügung, welches in der Fahrzeugindustrie als Kommunikationsprotokoll für Diagnosedienste zum Einsatz kommt und die ISO 14230 erfüllt. Es kann aber jedes andere Kommunikationsprotokoll eingesetzt werden, sofern es die nachfolgenden Aufgaben oder die ISO 14230 erfüllt.

Die Steuergeräte 1,3,5 weisen mindestens einen Mikro-Controller mit Prozessor-, Speicher- und Eingabe-/Ausgabe-Einheit zur Ausführung der Steuergerätefunktion, einen Kommunikations-Controller zur Umsetzung des Kommunikations-Protokolls und eine Senden-/Empfangseinheit zur Anbindung an den Datenbus 8 auf. Der Datenbus 8 ist als CAN-Datenbus mit entsprechender Protokollfunktionalität ausgebildet.

Die Software-Module 2,4,6,7 entsprechen softwaregesteuerten Applikationen, die auf dem jeweiligen Steuergerät 1,3,5 laufen. Die Steuergeräte 1,3,5 können mehrere Software-Module laden.

Die Steuergeräte 1;3;5 laden die im Speicher des Mikro-Controllers abgelegten Software-Module 2;4;6 in ihre Prozessoreinheit. Diese Software-Module 2;4;6 erfüllen die Hauptaufgaben des entsprechenden Steuergeräts 1;3;5. Das Software-Modul 7 kann zusätzlich von den Steuergeräten 1,3,5 geladen werden. Das Software-Modul 7 entspricht einer Nebenaufgabe der Steuergeräte

1,3,5. Das Software-Modul 7 ist ebenfalls im Speicher des Mikro-Controllers der Steuergeräte 1,3,5 abgelegt.

Beispielhaft übernimmt das Steuergerät 1 mittels des Software-Moduls 2 als Hauptaufgabe die Motorsteuerung, das Steuergerät 3 mittels des Software-Moduls 4 als Hauptaufgabe die Antriebsstrangssteuerung und das Steuergerät 5 mittels des Software-Moduls 6 als Hauptaufgabe die Steuerung des Bremsystems.

Das Software-Modul 7 übernimmt beispielhaft als Nebenaufgabe das Berechnen und Erstellen von Diagnosedaten, welche zur Anzeige im Verkehrsmittel 9 und/oder ablegen an zentraler Stelle im Verkehrsmittel 9 geeignet sind.

Das Software-Modul 7 kann in jedem Steuergerät 1,3,5 gestartet werden. Hierzu unterstützen die Steuergeräte 1,3,5 die Eingabe-/Ausgabeanforderungen des Software-Moduls 7. Beispielsweise werden Betriebsdaten von Sensoren oder Aktoren auf dem Datenbus 8 wie Öltemperatur, Stellmotorstellung usw. vom jeweiligen Steuergerät 1,3,5 als Daten an das Software-Modul 7 weitergeleitet.

Die vom Software-Modul 7 benötigte Prozesszeit entspricht der gesamten Zeit, in der das Software-Modul 7 einen bestimmten Prozessor von seinem Start bis zur Abarbeitung seiner Aufgabe in Anspruch genommen hat. Die Prozessorzeit ist insbesondere von der Taktfrequenz des im Mikro-Controller eines Steuergeräts 1,3,5 eingesetzten Prozessortyps abhängig.

Die Steuergeräte 1,3,5 arbeiten in Prozesszyklen, d.h. nach Ablauf einer bestimmten Zeit muss ein Prozesszyklus beendet sein und eine Ausgabe der im Prozess ermittelten Daten auf den Datenbus 8 erfolgen. Danach startet der Prozesszyklus erneut. Der Prozesszyklus der Steuergeräte 1;3;5 wird durch die Software-Module 2;4;6 der Hauptaufgabe und/oder Betriebssystem und/oder Busprotokoll bestimmt. Entsprechend werden die Prozesse, welche aus dem Laufen der Software-Module 2,4,6 auf dem

Prozessor des jeweiligen Mikro-Controllers des Steuergeräts 1,3,5 entstehen, als Hauptprozesse bezeichnet.

Die Steuergeräte 1,3,5 senden nach Ablauf eines Prozesszyklus bzw. einer Prozesszykluszeit Daten auf den Datenbus 8, welche ihre aktuelle Prozessorauslastung und eingesetzten Prozessortyp charakterisieren. Aus diesen Daten können die Steuergeräte 1,3,5 die Auslastung der weiteren Steuergeräte 1,3,5 ermitteln.

Die Auslastung eines Prozessors durch die Bearbeitung der Hauptaufgabe eines Steuergeräts 1,3,5 ist nicht gleichmäßig. Die Auslastung des Prozessors variiert je nach Anforderung der Hauptaufgabe. Beispielsweise ist die Prozessorauslastung des Steuergeräts 5 durch den Hauptprozess im Falle eines Bremseingriffes höher als ohne Bremseingriff. Ebenso ist die Prozessorauslastung des Steuergeräts 3 bei einem Schaltvorgang höher als ohne Schaltvorgang.

Das Software-Modul 7 kann auf den verschiedenen Steuergeräten 1,3,5 laufen. Die Entscheidung auf welchem der Steuergeräte 1,3,5 das Software-Modul 7 gestartet wird, hängt von der Rechenkapazität, also Prozessorauslastung und Prozessortyp, des jeweiligen Steuergeräts 1,3,5 ab.

Anhand des in Figur 2 dargestellten Flussdiagramms wird nun das erfindungsgemäße Verfahren erläutert, wobei im Folgenden angenommen wird, dass die Prozessoren der Steuergeräte 1,3,5 vom identischen Typ sind, also insbesondere dieselbe Taktfrequenz aufweisen und die Steuergeräte 1,3,5 sich im laufenden Betrieb befinden:

Prüfung 10:

Es wird geprüft, ob und auf welchem Steuergerät 1,3,5 das Software-Modul 7 läuft. Diese Prüfung muss turnusmäßig, also in bestimmten Zeitabschnitten erfolgen, da jedes der Steuergeräte 1,3,5 bei hoher Prozessorauslastung, das Software-Modul 7 ab-

schalten kann. Sobald die Abschaltung des Software-Moduls 7 erfolgt ist, muss das Software-Modul 7 wieder gestartet werden. Die Prüfung, ob und auf welchem Steuergerät 1,3,5 das Software-Modul 7 läuft, erfolgt, indem das Software-Modul 7 turnusmäßig oder auf Anfrage eine entsprechende Kennung, welche diese Daten enthält, auf den Datenbus 8 sendet.

Entscheidung 20:

Beispielhaft konnte eine entsprechende Kennung für das Software-Modul 7 im Schritt 10 auf dem Datenbus 8 nicht ermittelt werden, so dass in den Schritt 30 verzweigt werden muss.

Rechenkapazität 30:

Hierzu wird festgestellt, welches der beteiligten Steuergeräte 1,3,5 im Datenbus 8 die maximal freie Rechenkapazität, also die geringste Prozessorauslastung im Verhältnis zur Prozessor-Taktfrequenz, aufweist. Diese Information kann durch turnusmäßiges Senden seitens der beteiligten Steuergeräte 1,3,5 oder durch eine Abfrage erfolgen. Beispielhaft soll das Steuergerät 3 aktuell über die maximal freie Rechenkapazität verfügen.

Start Software-Modul 40:

Das Software-Modul 7 wird von dem im vorhergehenden Schritt 30 bestimmten Steuergerät 3 gestartet.

Software-Modul läuft 50:

Sobald das Software-Modul 7 ordnungsgemäß gestartet ist, sendet dieses turnusgemäß oder auf Anfrage eine Kennung über seinen Betriebszustand und sein Betriebssteuergerät, also auf welchem Steuergerät das Software-Modul 7 läuft, an den Datenbus 8.

Prüfung 10:

In der turnusgemäßen Prüfung wird ermittelt, ob und ggf. welche Kennung des Software-Moduls 7 auf dem Datenbus vorhanden ist.

Entscheidung 20:

Da der Schritt 10 ergibt, dass das Software-Modul 7 auf dem Steuergerät 3 läuft, ist zum Schritt 60 zu verzweigen.

Entscheidung 60:

Das Steuergerät 3 ermittelt seine eigene aktuelle Prozessorauslastung innerhalb eines Prozesszyklusses und vergleicht diese mit der aktuellen Rechenkapazität der weiteren Steuergeräte 1,2 innerhalb eines Prozesszyklusses. Hierzu fordert es die Information zur Rechenkapazität, also Prozessorauslastung und Prozessortyp, entweder von den Steuergeräten 1,2 an oder die Steuergeräte senden diese Informationen turnusgemäß auf den Datenbus 8.

Ist die Auslastung des Prozessors des Steuergeräts 3 im Vergleich zur Auslastung der Prozessoren der weiteren Steuergeräte 1,2 niedriger, so erfolgt keine Aktion. Das Software-Modul 7 läuft weiterhin auf dem Steuergerät 3. Die Verzweigung in den Prüfungsschritt 10 erfolgt turnusmäßig.

Ist die Auslastung des Prozessors des Steuergeräts 3 im Vergleich zur Auslastung der Prozessoren der weiteren Steuergeräte 1,2 höher, so erfolgt die Verzweigung zum Schritt 70.

Abschalten Software-Modul 70:

Es erfolgt die Abschaltung des Software-Moduls 7 im Steuergerät 3. Zudem ermittelt das Steuergerät 3 aus seinen Daten das Steuergerät 1,2 mit der aktuell maximal freien Rechenkapazität. Dies soll beispielhaft das Steuergerät 1 sein.

Start Software-Modul 40:

Das Software-Modul 7 wird von dem im vorhergehenden Schritt 70 bestimmten Steuergerät 1 gestartet.

Software-Modul läuft 50:

Sobald das Software-Modul 7 ordnungsgemäß gestartet ist, sendet dieses turnusgemäß oder auf Anfrage eine Kennung darüber, dass und auf welchem Steuergerät es läuft, an den Datenbus 8.

Es können auch mehrere, verschiedene Software-Module als Nebenaufgaben auf die Steuergeräte 1,3,5 verteilt werden. Zudem können die Steuergeräte 1,3,5 auch mehrere Hauptaufgaben erfüllen.

Die Umsetzung des erfindungsgemäßen Verfahrens erfolgt vorzugsweise auf Betriebssystemebene der Steuergeräte 1,3,5.

Der Datenbus 8 kann auch z.B. als FlexRay-Bus, als optischer MOST- oder D2B-Bus oder als elektrischer LIN-Bus in einem Verkehrsmittel, insbesondere einem Fahrzeug, vorgesehen sein.

Vorteilhafterweise ist das erfindungsgemäße Verfahren auch in sicherheitsrelevanten Systemen in Fahrzeugen einsetzbar. Diese Systeme werden, um die Ausfallsicherheit zu erhöhen, redundant ausgelegt, so dass bei Ausfall beispielsweise eines Steuergeräts auf ein redundant ausgelegtes Steuergerät umgeschaltet werden kann. Damit sind in redundant ausgelegten Systemen mehrere gleichartige Steuergeräte vorhanden, auf denen derselbe Hauptprozess läuft, nämlich die redundant ausgelegte Software-Applikation. Die notwendige Gleichartigkeit der redundant ausgelegten Steuergeräte impliziert, dass ein Software-Modul welches auf einem dieser redundant ausgelegten Steuergeräte lauffähig ist, auch auf den dazugehörigen weiteren zum redundant ausgelegten System gehörenden Steuergeräten lauffähig ist. Dies kann für die Anwendung des erfindungsgemäßen Verfahrens genutzt werden, indem nebengeordnete Software-Applikationen zusätzlich auf einem Steuergerät des redundanten Systems läuft.

Die Prozessorleistung der Steuergeräte 1,3,5 ist im bisher beschriebenen Verfahren derart ausgebildet, dass das Software-Modul 7 zur jeweiligen Hauptaufgabe des Steuergeräts 1,3,5 immer zugeschaltet werden kann, ohne dass der Hauptprozess auf Prozesszeit verzichten muss. Dieser Haupt-Prozess erhält also immer Vorrang vor allen anderen Prozessen, die auf dem Prozessor laufen. Sollte dem nicht so sein, muss im Schritt 60 und im Schritt 30 zusätzlich geprüft werden, ob die auf dem jeweiligen

Steuergerät 1,3,5 zur Verfügung stehende freie Rechenkapazität für die Bearbeitung der Nebenaufgabe ausreicht. Sollte dies nicht der Fall sein, kann das Software-Modul 7 in dem entsprechenden Steuergerät nicht gestartet werden. Für diese Berechnung muss die Prozesszeit für das Software-Modul 7 bei einem bestimmten Prozessortyp bereits vorab den Steuergeräten 1,3,5 bekannt sein.

Das erfindungsgemäße Verfahren kann ebenfalls angewendet werden, wenn die Prozessortypen in den Steuergeräten 1,3,5 unterschiedlich sind. Bei der Bestimmung der freien Rechenkapazität muss dann neben der Prozessorauslastung auch der Prozessortyp, also insbesondere die Prozessor-Taktfrequenz, berücksichtigt werden.

Das erfindungsgemäße Verfahren kann auch auf Steuergeräte ausgedehnt werden, deren Mikro-Controller mehrere Prozessoren aufweisen.

Das erfindungsgemäße Verfahren kann auch über ein zentrales Steuergerät gesteuert werden. Dies hat den Vorteil, dass das zentrale Steuergerät neben den Entscheidungs- und Berechnungsschritten 20,60,30 im Schritt 40 die entsprechende Software-Applikation an das Steuergerät verteilen kann.

DaimlerChrysler AG

Dr. Nili

02.12.2003

Patentansprüche

1. Verfahren zum Laden eines Software-Moduls in eine Prozessor-einheit eines Steuergeräts in einem Verkehrsmittel, wobei das Software-Modul (7) in mehreren Steuergeräten (1,3,5) lauffähig ist und die Steuergeräte (1,3,5) über einen Datenbus (8) Daten austauschen,
dadurch gekennzeichnet, dass
die Auswahl auf welchem Steuergerät (1,3,5) das Software-Modul (7) geladen wird, in Abhängigkeit der Rechenkapazität der sich aktuell im Betrieb befindlichen Steuergeräte (1,3,5) erfolgt.
2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass vor Ablauf des Software-Moduls (7) ermittelt wird, welches der weiteren Steuergeräte (1,3,5) die maximal freie Rechenkapazität zur Verfügung stellt und das Software-Modul (7) auf diesem Steuergerät (1;3;5) gestartet wird.
3. Verfahren nach Anspruch 1 bis 2, dadurch gekennzeichnet,
dass das Steuergerät (1;3;5) auf dem das Software-Modul (7) läuft seine Rechenkapazität mit der Rechenkapazität der weiteren Steuergeräte (1;3;5) vergleicht und in Abhängigkeit des Vergleichs das Software-Modul (7) beendet.
4. Verfahren nach Anspruch 1 oder 3, dadurch gekennzeichnet,
dass die Rechenkapazität der Steuergeräte (1,3,5) turnusmäßig oder auf Anfrage ermittelt wird.

5. Verfahren nach Anspruch 1 bis 4, dadurch gekennzeichnet, dass die Rechenkapazität eines Steuergeräts (1,3,5) aus der Prozessorauslastung und dem Prozessortyp ermittelt wird.
6. Verfahren nach Anspruch 1 bis 5, dadurch gekennzeichnet, dass das Software-Modul (7) auf dem Steuergerät (1,3,5) mit der maximalen freien Rechenkapazität gestartet wird.
7. Verfahren nach Anspruch 1 bis 6, dadurch gekennzeichnet, dass das Software-Modul (7) in den Speichermitteln der Steuergeräte (1,3,5) gespeichert ist.
8. Verfahren nach Anspruch 1 bis 7, dadurch gekennzeichnet, dass eine Kennung des Software-Moduls (7) turnusmäßig oder auf Anfrage auf den Datenbus (8) gesendet wird, wobei die Kennung Informationen über Betriebszustand und Betriebs-Steuergerät (1;3;5) des Software-Moduls (7) enthält.

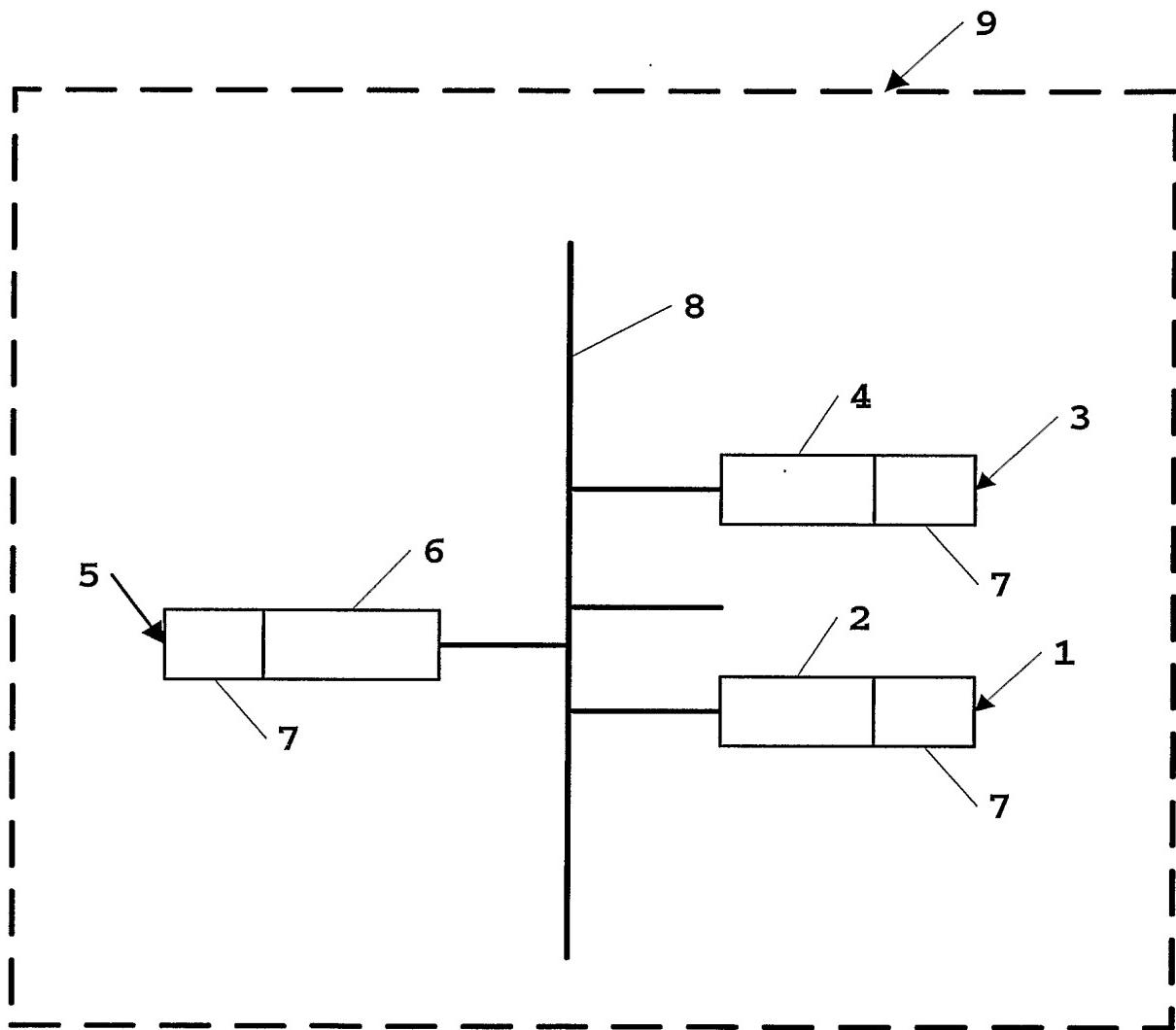
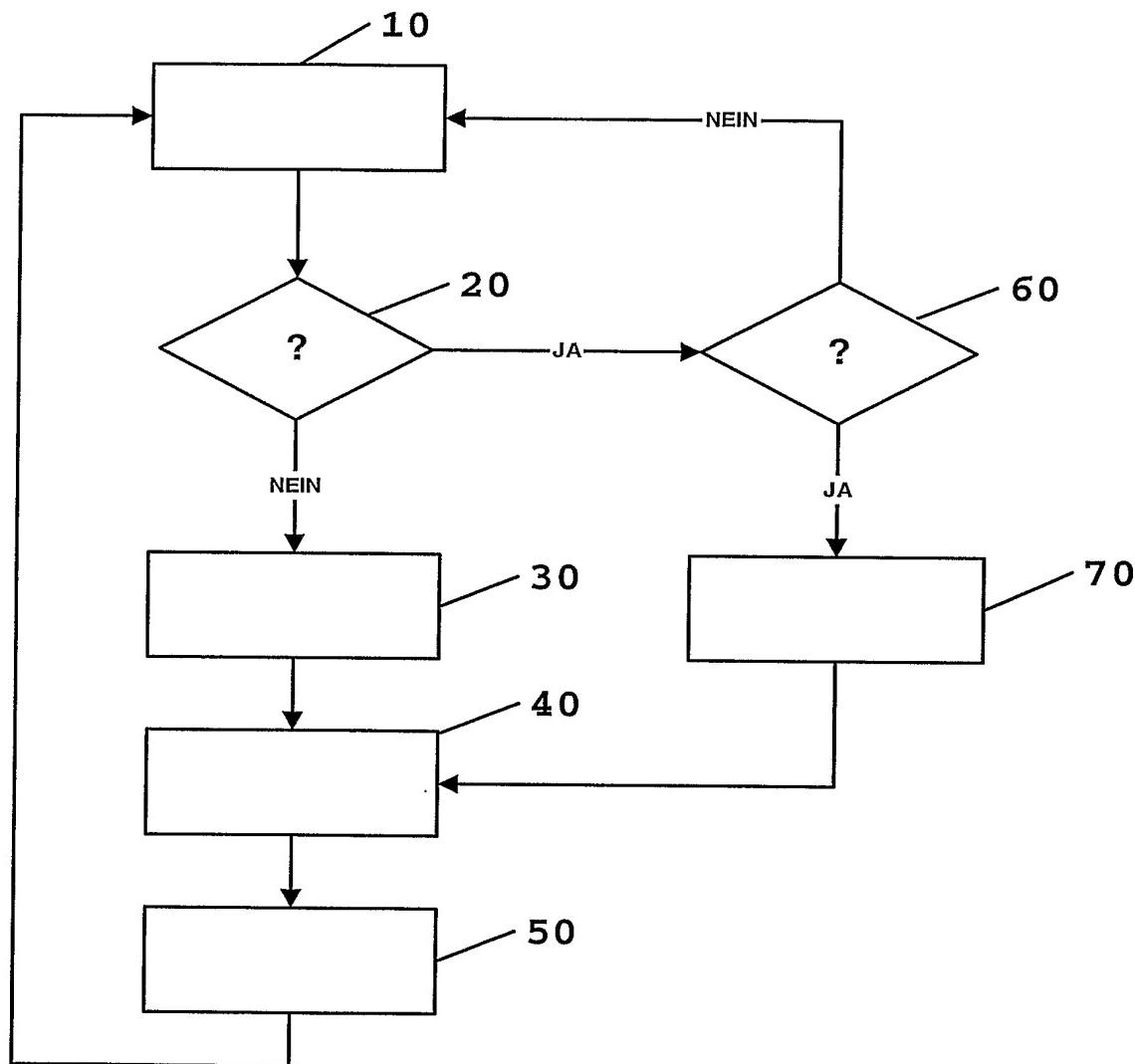


Fig. 1

**Fig. 2**

DaimlerChrysler AG

Dr. Nili

02.12.2003

Zusammenfassung

Die Erfindung betrifft ein Verfahren zum Laden eines Software-Moduls in eine Prozessoreinheit eines Steuergeräts in einem Verkehrsmittel. Es ist die Aufgabe der vorliegenden Erfindung die Prozessorauslastung in vernetzten Steuergeräten zu optimieren. Hierzu erfolgt die Auswahl auf welchem Steuergerät das Software-Modul geladen wird, in Abhängigkeit der Rechenkapazität der sich aktuell im Betrieb befindlichen Steuergeräte.

Figur 1

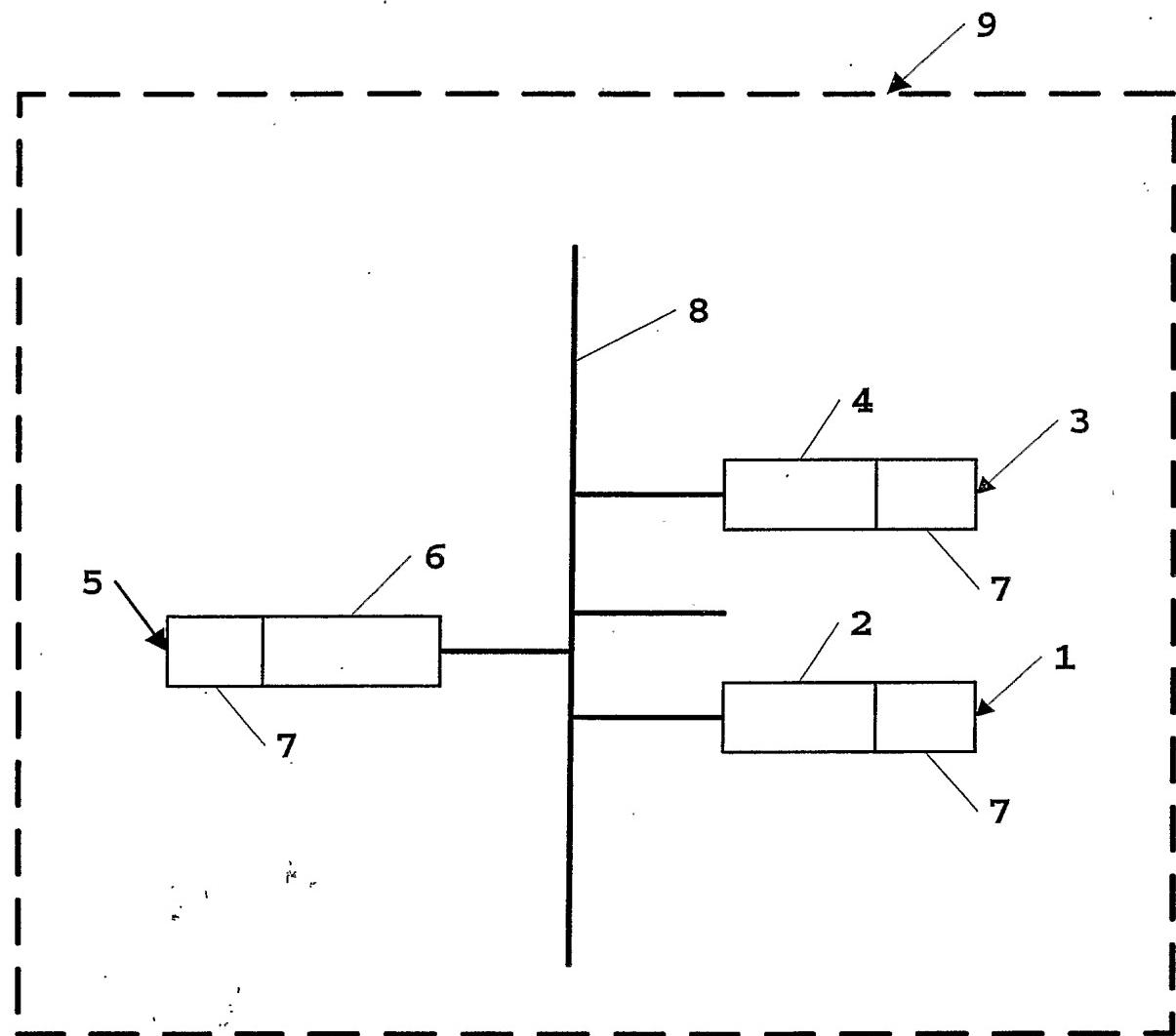


Fig. 1